Lecture 19

In this lecture we revisit the Towers of Hanoi puzzle, and then begin a more detailed study of logic.

Let's return to the Towers of Hanoi puzzle.

How many moves (1 disc at a time) are needed to get the stack of n discs from Pole 1 to Pole 3?

Let this number be f_n.

So $f_1 = 1$. (That is, if we only have 1 disc then a single move takes it to Pole 3.)

Let's look at the inductive proof that the puzzle can always be solved. When $n = k$, f_k moves are needed. So what happens when $n = k+1$? The number of moves needed is f_{k+1}, but this can be expressed in terms of f_k. It takes f_k moves to get the top k discs

to Pole 2, then 1 move to get the bottom disc to Pole 3, and then f_k moves to get the stack of k discs from Pole 2 and put it on top of the biggest disc which is now on Pole 3. So

$$f_{k+1} = f_k + 1 + f_k$$

i.e.,

$$f_{k+1} = 2 f_k + 1 .$$

This equation is an example of a <u>recurrence relation</u>, where each value of a function can be determined from previous values (assuming that we know some initial values).

Using n instead of k, we have

$$f_{n+1} = 2 f_n + 1$$

with <u>initial condition</u> $f_1 = 1$.

We can then construct the following partial table.

n	f_n	
1	1	
2	3	$\leftarrow f_2 = 2f_1 + 1 = 2+1 = 3$
3	7	$\leftarrow f_3 = 2f_2 + 1 = 6+1 = 7$
4	15	
5	31	etc.
6	63	

But suppose we want to know how many moves are needed to shift 100 discs. We would need to compute all the values from f_7 up to f_{99}, and then use

$$f_{100} = 2f_{99} + 1$$

to get the value of f_{100}. Clearly it would be better if we had a formula for f_n which only depended on n, rather than on earlier values of the function.

Such a formula is called a <u>solution</u> to the recurrence relation. We won't give a general method for solving recurrence relations, but rather we'll look at how to obtain a solution for this particular example.

Do the values in the second column of the table look familiar? If not, let's add 1 to every value and also put in some more rows.

n	f_n	$f_n + 1$
1	1	2
2	3	4
3	7	8
4	15	16
5	31	32
6	63	64
7	127	128
8	255	256
9	511	512
10	1023	1024

Now it should be clear that the new last column consists of all the powers of 2 with positive exponent. Specifically,

$$f_n + 1 = 2^n .$$

Subtracting 1 from both sides gives

$$f_n = 2^n - 1 .$$

It can be formally proved that this formula gives the correct answer. (We omit the details.)

Our reasoning here is an example of inductive reasoning, where we study enough examples to be able to hypothesize a general result.

Note that $f_{100} = 2^{100} - 1$, which is a rather big number!

Another look at logic.

Let P be a proposition (a statement declaring something to be true).

It has a truth value of T (for true) or F (for false).

Its negation is "not P", denoted by $\neg P$.

Here's the truth table for negation.

P	$\neg P$
T	F
F	T

Row 1 says, "When P is true, $\neg P$ is false."
" 2 " , "When " " false, " " true."

Let P and Q be propositions. They can be combined in various ways.

conjunction

$P \wedge Q$ means "P and Q"

(This says P is true <u>and</u> Q is true.)

disjunction

$P \vee Q$ means "P or Q"

(This says P is true <u>or</u> Q is true.)

implication

$P \Rightarrow Q$ means "P implies Q"

(This says <u>if</u> P is true <u>then</u> Q is true.)

equivalence

$P \Leftrightarrow Q$ means "P is equivalent to Q"

(This says P is true <u>if and only if</u> Q is true.)

Expressions created using the five <u>connectives</u>

$$\neg\ ,\ \wedge,\ \vee,\ \Rightarrow,\ \Leftrightarrow$$

are called <u>compound propositions.</u>

In the next lecture we'll look at the truth tables for all of these connectives.